COMPUTATIONAL THINKING IN VWO5
Een werkmiddag over 'Branch and Bound'

Mark Timmer en Joris van der Meulen

Beschrijving van de werkmiddag opdracht

De opdracht begint met het introduceren van planningsproblemen, waarbij een fabriek een machine heeft die verschillende taken moet verrichten. Iedere taak heeft een bepaalde *processing time* p_i (de tijd dat de machine met de taak bezig is), een *release date* r_i (het tijdstip waarop de taak beschikbaar is en de machine ermee aan de slag kan gaan) en een *due date* d_i (het moment waarop we zouden willen dat de taak klaar is). Voor een verzameling aan taken wordt deze informatie schematisch weergegeven zoals in tabel 1.

taak	1	2	3	4
p	4	2	6	5
r	0	1	3	5
d	8	12	11	10

tabel 1 Vier taken

De machine kan aan slechts één taak tegelijk werken en kan pas aan een taak beginnen zodra de release date van deze taak bereikt is. Ook kan de machine geen taken onderbreken. Merk overigens op dat het in dit voorbeeld niet mogelijk is om alle taken voor hun due date af te ronden. In de werkmiddag wordt de leerlingen ook gevraagd na te denken over hoe je dat sluitend zou kunnen aantonen – in dit geval kan dat betrekkelijk eenvoudig door vast te stellen dat de som van de processing times groter is dan het maximum van de due dates.
Leerlingen puzzelen vervolgens wat met de volgorde van deze taken, om gevoel te krijgen voor de problematiek. Vervolgens wordt gesproken over een methode om verschillende volgordes met elkaar te vergelijken. Aan de hand van een praktisch voorbeeld van een bedrijf, waarbij klanten begrip op kunnen brengen voor een beetje vertraging maar naar de concurrent gaan als hun product te lang na de deadline af is, komen leerlingen (door na te denken over een aantal aangedragen opties) bij een criterium dat oplossingen beoordeelt op basis van de *maximale vertraging*. De vertraging van een taak is hierbij gedefinieerd als 0 als deze op tijd of te vroeg is afgerond, en anders gedefinieerd als het verschil tussen de due date en de daadwerkelijke afrondingstijd.
Vervolgens worden de leerlingen tot nadenken over computationele complexiteit aangezet, door te redeneren over het totaal aantal volgordes dat er is – aangezien ze bij Wiskunde B geen combinatoriek hebben gehad is *faculteit* leerlingen nog onbekend, wat dit een uitdagende opgave maakt. Door vervolgens te berekenen hoe lang complete enumeratie zou duren bij twintig taken, als de computer vijftig nanoseconden nodig heeft per mogelijke oplossing, krijgen ze een gevoel voor de dramatische effecten van dergelijke groei. Ook wordt hen gevraagd om na te denken over heuristieken om tot een potentieel goede oplossing te komen. Een redelijk werkende heuristiek voor dit type problemen is als volgt:
1 Beschouw de taken die nog niet uitgevoerd zijn en waarvan het lanceermoment al bereikt is, oftewel: de taken waar je nu aan zou kunnen beginnen. (Als van geen enkele taak die nog uitgevoerd moet worden het lanceermoment al bereikt is, wacht dan tot dit wel het geval is).
2 Kies uit deze beschikbare taken degene met het vroegste vervalmoment (oftewel: de taak die het eerst af moet zijn) en voer die uit. Als dit een keus uit meerdere taken oplevert (omdat meerdere taken op hetzelfde moment af moeten zijn), kies daar dan willekeurig eentje van.
3 Herhaal de bovenstaande twee stappen totdat alle taken klaar zijn.
Voor het probleem uit tabel 1 levert deze heuristiek inderdaad de optimale oplossing. Leerlingen worden daarna uitgedaagd om aan te tonen dat deze heuristiek niet altijd de juiste oplossing oplevert, door de getallen in tabel 1 dusdanig aan te passen dat voor de nieuwe probleemsituatie de heuristiek een suboptimale oplossing geeft – dit bleek voor velen nog niet zo eenvoudig, maar is naar ons idee wel een erg interessante activiteit. Een mogelijke oplossing hiervoor is weergegeven in tabel 2.

i	1	2	3	4
p	5	5	5	5
r	1	0	0	0
d	6	30	30	30

tabel 2

De heuristische aanpak gebiedt ons nu om met een van de drie taken te beginnen die direct beschikbaar zijn, waardoor taak 1 te laat klaar zal zijn. Verstandiger zou het natuurlijk zijn geweest om één tijdseenheid te wachten, dan eerst taak 1 uit te voeren en daarna de andere drie taken te voltooien – op die manier zijn alle taken op tijd klaar en is de oplossing dus optimaal.
Vervolgens wordt branch and bound erbij gehaald, om efficiënter aan dit type problemen te kunnen rekenen. De 'combinatorische explosie' aan mogelijkheden en het feit dat de bedachte heuristieken blijkbaar niet altijd optimale oplossingen leveren, maakt de noodzaak voor een alternatieve aanpak duidelijk. Leerlingen krijgen hiertoe eerst een gedeelte te zien van een figuur zoals de eerdere afbeeldingen hierboven van boomstructuren waarin de mogelijke oplossingen weergegeven worden. In dit geval komen representaties van oplossingen voor zoals 13✶✶, om aan te duiden dat we beginnen met taak 1, vervolgens taak 3 uitvoeren en dat de laatste twee taken nog niet vastliggen.
Het bepalen van ondergrenzen is in deze context wat ingewikkelder dan bij het toewijzingsprobleem van eerder. Een nuttige observatie hierbij is dat, als alle taken direct beschikbaar zouden zijn en we nog steeds uitgaan van het minimaliseren van de maximale vertraging, we altijd het best kunnen beginnen met de taak die het eerst klaar moet zijn (*earliest due date first*). Dat idee kan vervolgens worden uitgebouwd tot de situatie waarin niet alle taken direct beschikbaar zijn, waarbij we bovendien aannemen dat taken onderbroken en hervat kunnen worden (*pre-emption*). Als dat het geval zou zijn, dan geeft de eerder uitgelegde heuristiek (uit de taken die beschikbaar zijn steeds degene uitkiezen die het eerst klaar moet zijn) de optimale oplossing als hij wordt aangevuld met het idee om de huidige taak af te breken als er een nieuwe taak beschikbaar komt die eerder klaar moet zijn. Hoewel dat afbreken in werkelijkheid niet mag, geeft het wel een goede ondergrens om bij branch and bound te gebruiken. Dit inzicht is voor leerlingen nog niet zo eenvoudig en vereist soms wat extra mondelinge toelichting.

Leerlingen oefenen vervolgens met pre-emption en gebruiken dit om een ondergrens te bepalen voor het probleem van de vier taken van tabel 1. We dragen aan dat bij een situatie zoals 12✶✶ de ondergrens berekend kan worden door (indien nodig) pre-emption toe te passen op de laatste twee taken. Vervolgens moeten leerlingen zelf nadenken over hoe dit toegepast kan worden in de boomstructuur. Ze moeten zelf nadenken over wanneer je nu zeker weet dat je klaar bent en de optimale oplossing gevonden hebt. Wie aan de slag gaat met dit probleem zal uiteindelijk uitkomen op de optimale planning 1-3-4-2.

Daarna worden nog enkele opgaven gemaakt waarin inzicht in de techniek branch and bound getoetst wordt, bijvoorbeeld door bij een gegeven 'tussenstand' van de berekening te laten beredeneren wat de volgende stap is én wat de voorgaande stap is geweest. Als laatste oefenen leerlingen nog met het toepassen van branch and bound op een voor hen nieuw probleem, namelijk de toekenning van mensen aan taken zoals aan het begin van dit artikel besproken is. Daarbij moeten ze dan ook zelf nadenken over de bepaling van ondergrenzen en de manier van branchen.